

Python - Basics

1 Introduction to Python

- Python is a high-level, interpreted, general-purpose programming language.
- It was created by Guido van Rossum and first released in 1991.
- Python is known for its simplicity and readability.

Features of Python

- Easy to learn and use
- High-level language
- Interpreted language
- Portable (cross-platform)
- Extensive standard library
- Supports multiple programming paradigms
- Open source and free

2 Executing a Simple "Hello World" Program

(A) Interactive Mode

Steps:

- Open Python interpreter (IDLE).
- Type the following command and press Enter.

```
>>> print("Hello World")
```

Hello World

(B) Script Mode

Steps:

- Open IDLE → File → New File
- Write the program:

```
print("Hello World")
```
- Save the file (e.g., hello.py)
- Run the program (F5) to see the output.

Remember!

Interactive Mode is used for quick testing.
Script Mode is used for writing and running complete programs.

3 Python Character Set

Python uses the following character set:

- Letters: a to z, A to Z
- Digits: 0 to 9
- Special Characters: + - * / % = () [] { } , . : ; ' " < > ? @ # \$ _ & | ~ ^ ` \
- Whitespace Characters: space, tab, newline

Exam Tip

Python is case-sensitive.
So, variable and Variable are different.

4 Python Tokens

Tokens are the smallest individual units in a Python program.

Token Type	Description	Example
1. Keyword	Reserved words in Python that have special meaning.	if, else, for, while, def, True, False, None
2. Identifier	Names given to variables, functions, classes, etc.	x, total, my_list, calculateSum
3. Literal	Fixed values in the program.	10, 3.14, 'Hello', True
4. Operator	Symbols that perform operations.	+, -, *, /, ==, and, or
5. Punctuator	Symbols used for structure and separation.	(), ; , ' " [] { }

Examples in a Statement

```
x = 10 + 5
```

↑ ↑ ↑ ↑ ↑
Identifier Operator Literal

5 Variables

- A variable is a name given to a memory location to store data.
- In Python, you do not need to declare the type of variable.
- Syntax: variable_name = value

Rules for Naming Variables

- Must start with a letter (A-Z or a-z) or underscore (_).
- Can contain letters, digits and underscores.
- Cannot start with a digit.
- Case-sensitive (age, Age and AGE are different).
- Cannot be a Python keyword.

Examples

- ✓ age
- ✓ total_marks
- ✓ _value1
- ✗ 1stvalue
- ✗ total-marks
- ✗ for

6 L-value and R-value

- L-value (Left-value)**: Appears on the left side of assignment operator. It represents a memory location.
- R-value (Right-value)**: Appears on the right side of assignment operator. It represents the value stored in a memory location.

Example

```
x = 25
```

↑ ↑
L-value R-value
(variable) (value)

7 Operators in Python

(A) Arithmetic Operators

- + Addition
- Subtraction
- * Multiplication
- / Division
- // Floor Division
- % Modulus
- ** Exponentiation

(B) Relational Operators

- = Equal to
- != Not Equal to
- > Greater than
- < Less than
- >= Greater than or equal to
- <= Less than or equal to

(C) Logical Operators

- and True if both conditions are True
- or True if at least one condition is True
- not Reverses the result (True → False, False → True)

(D) Assignment Operators

- = Assign
- += Add and Assign
- = Subtract and Assign
- *= Multiply and Assign
- /= Divide and Assign
- %= Modulus and Assign

(E) Augmented Assignment Operators

Same as (D)
But shorter way

Example:

`x += 5` → `x = x + 5`

(F) Identity Operators

- is Returns True if both operands are the same object
- is not Returns True if both operands are not the same object

(G) Membership Operators

- in Returns True if a value is found in the sequence
- not in Returns True if a value is not found in the sequence

Quick Example

- 5 in [1, 2, 3, 4, 5] → True
- 10 not in [1, 2, 3] → True